

EXERCISE 03

SIMPLE MACROS

Purpose of classes

Implementation of simple macros using Range class objects and assignment statement.

Materials to be prepared

Materials placed on the course website:

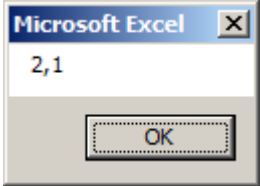
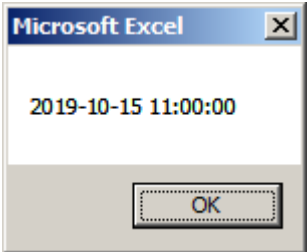
- Exercise01
- Lecture01.

Tasks

Write the macros described below.

Note! Macros will be written in two versions: first without using variables, second with the use of variables.

1.	Macro length_1 calculates the length of the vector $v = (x, y)$, i.e.: $\ v\ = \sqrt{x^2 + y^2}$. It is assumed that the coordinates of the vector are in cells A1, B1 and the calculated length of the vector should be written in cell C1 (see Note 1).
2.	Macro positive converts the number from the active cell into the corresponding positive number (see Note2).
3	Macro fractional calculates the fractional part of the number from the active cell. Regardless of the sign of the number, the fractional part should be calculated as positive. Result (fractional part) should be written in the cell to the right of the active cell (see Note3).
4	Macro add_hour , assuming that date and time are entered in the active cell, increases the value in the cell by one hour (see Note 4).
5	Macro add_hours1 increases the value in the active cell (assuming that it contains a date and time) by the number of hours given in the adjacent cell.
6	Macro add_hours2 increases the value in the active cell (assuming that it contains a date and time) by the number of hours in cell A1.
7	Macro sum_1 calculates the sum of the selected block of cells (Selection) and displays the message <i>"The sum of the values in the selected range is : □□□"</i> □□□ is the calculated sum (see Note 5).
8	Macro sum_2 calculates the sum of the squares of the selected block of cells and displays the message <i>"The sum of the squares of the values in the selected range is : □□□"</i> □□□ is the calculated sum (see Note6).
9	Macro gap calculates the range (i.e. the difference between the maximum and minimum values) for the selected block of cells and displays the message <i>"The range of values in the specified range is: □□□"</i> □□□ is the calculated range (see Note7).

Notes	
1	In your calculations, use the exponentiation operator and/or the square root function (see lecture, slide Arithmetic operations)
2	A negative number can be converted to a positive number using the function: <pre>Abs(number As Double) As Double</pre> example: <code>MsgBox Abs(-2.1)</code> <code>MsgBox Abs(2.1)</code> ' applying the function for a positive number gives the same result 
3	The fractional part of a number can be determined by subtracting its integer part from the number. The integer part of a number can be retrieved using the functions: <pre>Int(number As Double) As Double</pre> <pre>Fix(number As Double) As Double</pre> number is the number whose integer part will be determined. The difference between functions is only visible for negative numbers. The <code>Int</code> function returns an integer smaller than the given one and <code>Fix</code> returns a larger number, e.g. <pre>Int(-2.1) ' returns -3</pre> <pre>Fix(-2.1) ' returns -2.</pre>
4	A date/time variable can be enlarged using the function: <pre>DateAdd(interval As String, number As Double, date As Variant) _ As Variant</pre> interval represents the unit of time by which the date will be increased or decreased, in the case of modifying hours, use "h", number is a positive or negative number specifying the number of time units date represents the date to be modified  example: <code>MsgBox DateAdd("h", 1, #10/15/2019 10:00:00 AM#)</code>

5

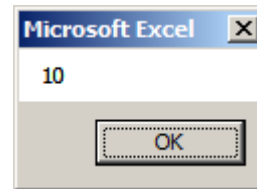
The sum of the given numbers and/or ranges can be calculated using the **worksheet function**:

```
Sum(z1 As Variant, ..., z30 as Variant) As Double
z1 ... z30 are numbers or ranges to sum
```

Worksheet functions are available as methods of the WorksheetFunction object provided by the Application object.

example:

	A	B
1	1	3
2	2	4



```
MsgBox Application.WorksheetFunction.Sum(Range("A1:B2"))
```

6

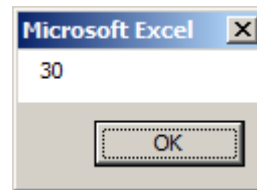
The sum of squares of given numbers and/or ranges can be calculated using **worksheet function**:

```
SumSq(z1 As Variant, ..., z30 as Variant) As Double
z1 ... z30 are numbers or ranges to sum
```

Worksheet functions are provided by the WorksheetFunction object of the Application object.

example:

	A	B
1	1	3
2	2	4



```
MsgBox Application.WorksheetFunction.SumSq(Range("A1:B2"))
```

7

The minimum and maximum of the given numbers and/or ranges can be calculated using **worksheet functions**:

```
Min(z1 As Variant, ..., z30 as Variant) As Double
```

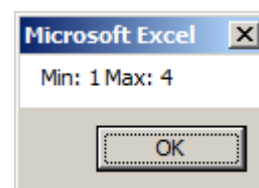
```
Max(z1 As Variant, ..., z30 as Variant) As Double
```

z1 ... z30 are the numbers or ranges being searched

Worksheet functions are provided by the WorksheetFunction object of the Application object.

przykład:

	A	B
1	1	3
2	2	4



```
MsgBox "Min:" & Application.WorksheetFunction.Min(Range("A1:B2")) _
    & " Max:" & Application.WorksheetFunction.Max(Range("A1:B2"))
```